Supplementary

1. More Ablation Analysis

In this section, we conduct more ablation studies on multiple parameters that are used in Algorithm 1 and 2.

1.1. Visualization of Partial/Occluded Objects

As we mentioned in the main text, our method is robust in detecting partial/occluded objects. Here we visualize several objects of different partial indexes in Figure 1. We can see that small partial indexes indicate occluded objects, while large partial indexes indicate complete objects. Our method gives higher average recall on partial objects compared with previous methods.

Figure 1. Point clouds of chairs with different partial indexes in ScanNet dataset. We see that small partial indexes refer to objects are occluded and partial, while objects with large partial indexes are more complete.

1.2. Effects of δ in Bounding Box Generation

In Algorithm 2, we iteratively generate bounding boxes whenever the maximum value of the heatmap is above a threshold δ. Intuitively, with a small threshold, more object candidates are detected but may degrade the performance by introducing false positives at the same time. With a large threshold, our model is more confident on the detected bounding boxes, but may lower the recall by missing small objects.

Here we investigate how this threshold influences the final mAP_{50} on ScanNet val dataset. quantitative results are given in Figure 2. The optimal threshold for δ lies around 60, with a balance between recall and precision.

1.3. Effects of K in Canonical Voting Process

We report our joint model's average processing time per scan and mAP_{50} for different values of K of Algorithm 1 in Figure 3. We see as K increases, the processing time arises linearly because of the exhaustive search; while the mAP gets saturated after $K = 120$.

2. Extension to Full 3D Rotations

Though we mainly conduct experiments on common indoor scenes, which usually contains only 1D rotations around the gravity axis, our voting scheme can be extended to full 3D rotation and be applied to NOCS REAL275 dataset (a 6D pose estimation benchmark). Specifically, we now treat every pixel in a given RGB-D frame as a 3D point which generates center offsets over $\frac{4\pi}{K}$ solid angles (K is resolution) instead of $\frac{2\pi}{K}$ circle angles. After candidate centers have been proposed, we collect all the points that contribute to these centers within some radius tolerance. The LCCs of these points are then used with the Umeyama algorithm to solve 3D rotations in closed form. Interestingly, our voting scheme on NOCS REAL275 achieves **17.0**, **40.7** and **45.8** mAP for (5°, 5 cm), (10°, 5 cm) and (10°, 10 cm) metrics respectively, beating NOCS [2] baseline by a large margin.

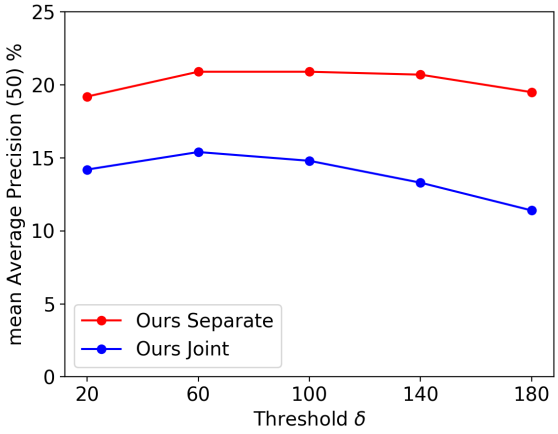

Figure 2. **mAP results on ScanNet val dataset under different δ.**

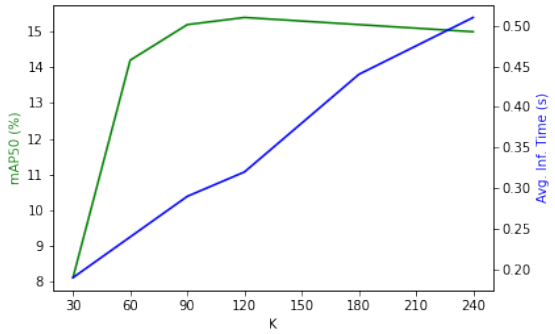

Figure 3. Average processing time and mAP_{50} of our joint model on ScanNet. $K = 120$ keeps a balance between time and accuracy.

3. Details of Vote Map Guided Refinement Module on SUN RGB-D dataset

For SUN RGB-D dataset, due to the lack of symmetric information and limited training data, instead of a determinis-

tic bounding box generation procedure, we sample multiple box candidates with probability proportional to the vote map, and then leverage a refinement module to further refine these bounding boxes. The pipeline is shown in Figure 4. The vote map generation is exactly the same as that in Algorithm 1. Nevertheless, instead of direct bounding box generation with back-projection checking, we use a neural refinement module with learnable parameters to generate final bounding boxes. Specifically, we first generate 512 possible object center candidates according to the 3D vote map. The vote map is normalized to form a proper distribution. Then these vote proposals are then passed through a proposal refinement and classification module that is similar to BRNet. Every object proposal is aggregated with its surrounding back-traced representative points, using max pooling on their high dimensional embeddings generated by PointNet++. For more details of this refinement module, we refer the reader to BRNet [1].

References

[1] Bowen Cheng, Lu Sheng, Shaoshuai Shi, Ming Yang, and Dong Xu. Back-tracing representative points for voting-based 3d object detection in point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8963–8972, 2021. 2

[2] He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, and Leonidas J Guibas. Normalized object coordinate space for category-level 6d object pose and size estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2642–2651, 2019. 1

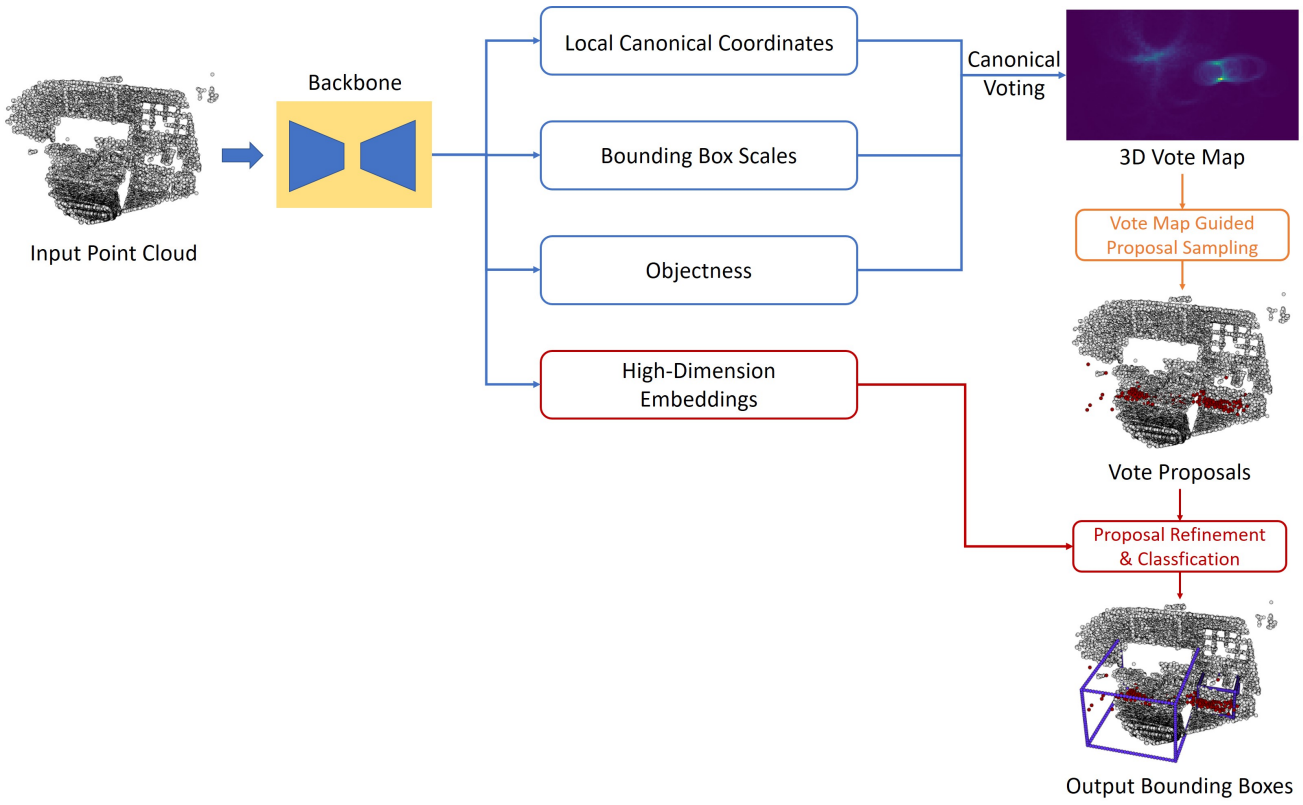

Figure 4. **The architecture of our method on SUN RGB-D dataset.**